This listing of claims replaces all prior versions, and listings of claims in the instant application:

Listing of Claims:

1. (Currently Amended) A method for dynamic implementation of a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing, dynamically and automatically at runtime, a package proxy JMI interface when said request comprises a package proxy request;

implementing, dynamically and automatically at runtime, a class proxy JMI interface when said request comprises a class proxy request; and

implementing, dynamically and automatically at runtime, a class instance JMI interface when said request comprises a class instance request wherein said implementing operations are performed separately and independently based upon said request.

2. (Original) The method of claim 1 wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface;

creating a new instance of said class; and

returning said instance.

3. (Currently Amended) ~~The method of claim 2 wherein said generating further comprises:~~ A method for dynamic

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

implementation of a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a package proxy JMI interface when said request comprises a package proxy request wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface wherein said generating further comprises:

receiving a metamodel package;

receiving a package proxy interface method associated with said metamodel package;

determining a class name based upon said interface method;

searching said metamodel package for a class corresponding to said class name; and

producing an implementation of said interface method that returns a proxy for said class when said class name is found in said metamodel package;

creating a new instance of said class; and

returning said instance;

implementing a class proxy JMI interface when said request comprises a class proxy request; and

implementing a class instance JMI interface when said request comprises a class instance request.

4. (Previously Presented) The method of claim 3 wherein said implementation of said interface method calls a handler

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

method of a superclass of said class, passing said class name as an argument and returning the proxy for said class.

5. (Original) The method of claim 1 wherein said implementing a class proxy JMI interface comprises:

generating bytecode for a class that implements said class proxy JMI interface;

creating a new instance of said class; and

returning said instance.

6. (Currently Amended) ~~The method of claim 5~~ A method for dynamic implementation of a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a package proxy JMI interface when said request comprises a package proxy request;

implementing a class proxy JMI interface when said request comprises a class proxy request wherein said implementing a class proxy JMI interface comprises:

generating bytecode for a class that implements said class proxy JMI interface wherein said generating further comprises:

receiving a metamodel class;

receiving a class proxy interface method associated with said metamodel class;

producing a first implementation of said interface method that creates a new instance of said class when said interface method is parameterless; and

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

producing a second implementation of said interface method that creates a new instance of said class and sets attributes passed as arguments to said interface method when said interface method includes at least one parameter;

creating a new instance of said class; and

returning said instance; and

implementing a class instance JMI interface when said request comprises a class instance request wherein said implementing operations are performed separately and independently based upon said request.

7. (Previously Presented) The method of claim 6 wherein said first implementation calls a handler method of a superclass of said class, passing said class name as an argument and returning a new instance of said class.

8. (Previously Presented) The method of claim 6 wherein said second implementation calls a handler method of a superclass of said class, passing said class name, said attributes, and attribute values as arguments and returning a new instance of said class.

9. (Original) The method of claim 1 wherein said implementing a class instance JMI interface comprises:

generating bytecode for a class that implements said class instance JMI interface;

creating a new instance of said class; and

returning said instance.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

10. (Currently Amended) ~~The method of claim 9~~ A method for dynamic implementation of a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a package proxy JMI interface when said request comprises a package proxy request;

implementing a class proxy JMI interface when said request comprises a class proxy request; and

implementing a class instance JMI interface when said request comprises a class instance request wherein said implementing a class instance JMI interface comprises:

generating bytecode for a class that implements said class instance JMI interface wherein said generating further comprises:

receiving a metamodel class;

receiving a class instance interface method associated with said metamodel class, said interface method having an interface method name;

producing a first implementation of said interface method that sets the value of an attribute when said interface method name includes a first prefix and when the attribute associated with said interface method is found in said metamodel class;

producing a second implementation of said interface method that sets the value of a reference when said interface method name includes a first prefix and when the reference

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

associated with said interface method is found in said metamodel class;

producing a third implementation of said interface method that gets the value of an attribute when said interface method name includes a second prefix and when the attribute associated with said interface method is found in said metamodel class;

producing a fourth implementation of said interface method that gets the value of a reference when said interface method name includes a second prefix and when the reference associated with said interface method is found in said metamodel class; and

producing a fifth implementation of said interface method that executes an operation when said interface method has the same name as said operation; and

creating a new instance of said class; and

returning said instance.

11. (Original) The method of claim 10 wherein

said first prefix is "set"; and

said second prefix is "get".

12. (Previously Presented) The method of claim 10 wherein said producing a first implementation further comprises:

receiving an attribute name and an attribute value; and

producing an implementation that calls a handler method of a superclass of said class, passing said attribute name and said attribute value as arguments.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

13. (Previously Presented) The method of claim 10 wherein said producing a second implementation further comprises:

receiving a reference name and an reference value; and

producing an implementation that calls a handler method of a superclass of said class, passing said reference name and said reference value as arguments.

14. (Previously Presented) The method of claim 10 wherein said producing a third implementation further comprises:

receiving an attribute name;

producing an implementation that calls a handler method of a superclass of said class, passing said attribute name as an argument and returning the attribute value associated with said attribute name; and

returning said attribute value.

15. (Previously Presented) The method of claim 10 wherein said producing a fourth implementation further comprises:

receiving a reference name;

producing an implementation that calls a handler method of a superclass of said class, passing said reference name as an argument and returning the reference value associated with said reference name; and

returning said reference value.

16. (Previously Presented) The method of claim 10 wherein said producing a fifth implementation further comprises:

receiving an operation name and any associated arguments;

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

producing an implementation that calls a handler method of a superclass of said class, passing said operation name and said associated arguments as arguments and returning an operation return value; and

returning said operation return value.

17. (Currently Amended) A method for dynamic implementation of a Java™ Metadata Interface (JMI), the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing, dymanically and automatically, a JMI interface when said JMI interface is unimplemented; and

executing a stored JMI interface implementation when said JMI interface is implemented.

18. (Original) The method of claim 17 wherein said implementing further comprises:

implementing a package proxy JMI interface when said request comprises a package proxy request and when said package proxy JMI interface is unimplemented;

implementing a class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is unimplemented; and

implementing a class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is unimplemented; and

said executing further comprises:

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

executing a stored a package proxy JMI interface implementation when said request comprises a package proxy request and when said package proxy JMI interface is implemented;

executing a stored class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is implemented; and

executing a stored class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is implemented.

19. (Original) The method of claim 18 wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface;

creating a new instance of said class; and

returning said instance.

20. (Currently Amended) ~~The method of claim 19~~ A method for dynamic implementation of a Java™ Metadata Interface (JMI), the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a JMI interface when said JMI interface is unimplemented, wherein said implementing further comprises:

implementing a package proxy JMI interface when said request comprises a package proxy request and when said package proxy JMI interface is

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

unimplemented wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface wherein said generating further comprises:

receiving a metamodel package;

receiving a package proxy interface method associated with said metamodel package;

determining a class name based upon said interface method;

searching said metamodel package for a class corresponding to said class name; and

producing an implementation of said interface method that returns a proxy for said class when said class name is found in said metamodel package;

creating a new instance of said class; and

returning said instance;

implementing a class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is unimplemented; and

implementing a class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is unimplemented; and

executing a stored JMI interface implementation when said JMI interface is implemented wherein said executing further comprises:

executing a stored a package proxy JMI interface implementation when said request comprises a package proxy request and when said package proxy JMI interface is implemented;

executing a stored class proxy JMI interface when said request comprises a class proxy request

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

and when said class proxy JMI interface is implemented; and

executing a stored class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is implemented.

21. (Previously Presented) The method of claim 20 wherein said implementation of said interface method calls a handler method of a superclass of said class, passing said class name as an argument and returning the proxy for said class.

22. (Original) The method of claim 18 wherein said implementing a class proxy JMI interface comprises:

generating bytecode for a class that implements said class proxy JMI interface;

creating a new instance of said class; and

returning said instance.

23. (Currently Amended) ~~The method of claim 22~~ A method for dynamic implementation of a Java™ Metadata Interface (JMI), the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a JMI interface when said JMI interface is unimplemented, wherein said implementing further comprises:

implementing a package proxy JMI interface when said request comprises a package proxy request and

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

when said package proxy JMI interface is unimplemented;

implementing a class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is unimplemented wherein said implementing a class proxy JMI interface comprises:

generating bytecode for a class that implements said class proxy JMI interfacewherein said generating further comprises:

receiving a metamodel class;

receiving a class proxy interface method associated with said metamodel class;

producing a first implementation of said interface method that creates a new instance of said class when said interface method is parameterless; and

producing a second implementation of said interface method that creates a new instance of said class and sets attributes passed as arguments to said interface method when said interface method includes at least one parameter;

creating a new instance of said class; and

returning said instance; and

implementing a class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is unimplemented; and

executing a stored JMI interface implementation when said JMI interface is implemented wherein said executing further comprises:

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

executing a stored a package proxy JMI interface implementation when said request comprises a package proxy request and when said package proxy JMI interface is implemented;

executing a stored class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is implemented; and

executing a stored class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is implemented.

24. (Previously Presented) The method of claim 23 wherein said first implementation calls a handler method of a superclass of said class, passing said class name as an argument and returning a new instance of said class.

25. (Previously Presented) The method of claim 23 wherein said second implementation calls a handler method of a superclass of said class, passing said class name, said attributes, and attribute values as arguments and returning a new instance of said class.

26. (Original) The method of claim 18 wherein said implementing a class instance JMI interface comprises:

generating bytecode for a class that implements said class instance JMI interface;

creating a new instance of said class; and

returning said instance.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

27. (Currently Amended) ~~The method of claim 26~~ A method for dynamic implementation of a Java™ Metadata Interface (JMI), the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a JMI interface when said JMI interface is unimplemented, wherein said implementing further comprises:

implementing a package proxy JMI interface when said request comprises a package proxy request and when said package proxy JMI interface is unimplemented;

implementing a class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is unimplemented; and

implementing a class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is unimplemented wherein said implementing a class instance JMI interface comprises:

generating bytecode for a class that implements said class instance JMI interface wherein said generating further comprises:

receiving a metamodel class;

receiving a class instance interface method associated with said metamodel class, said interface method having an interface method name;

producing a first implementation of said interface method that sets the value of an attribute when said interface method

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

name includes a first prefix and when the attribute associated with said interface method is found in said metamodel class;

producing a second implementation of said interface method that sets the value of a reference when said interface method name includes a first prefix and when the reference associated with said interface method is found in said metamodel class;

producing a third implementation of said interface method that gets the value of an attribute when said interface method name includes a second prefix and when the attribute associated with said interface method is found in said metamodel class;

producing a fourth implementation of said interface method that gets the value of a reference when said interface method name includes a second prefix and when the reference associated with said interface method is found in said metamodel class; and

producing a fifth implementation of said interface method that executes an operation when said interface method has the same name as said operation;

creating a new instance of said class; and

returning said instance; and

executing a stored JMI interface implementation when said JMI interface is implemented wherein said executing further comprises:

executing a stored a package proxy JMI interface implementation when said request comprises a package

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

proxy request and when said package proxy JMI interface is implemented;

executing a stored class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is implemented; and

executing a stored class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is implemented.

.

28. (Original) The method of claim 27 wherein
said first prefix is "set"; and
said second prefix is "get".

29. (Previously Presented) The method of claim 27 wherein said producing a first implementation further comprises:

receiving an attribute name and an attribute value; and

producing an implementation that calls a handler method of a superclass of said class, passing said attribute name and said attribute value as arguments.

30. (Previously Presented) The method of claim 27 wherein said producing a second implementation further comprises:

receiving a reference name and an reference value; and

producing an implementation that calls a handler method of a superclass of said class, passing said reference name and said reference value as arguments.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

31. (Previously Presented) The method of claim 27 wherein said producing a third implementation further comprises:

receiving an attribute name;

producing an implementation that calls a handler method of a superclass of said class, passing said attribute name as an argument and returning the attribute value associated with said attribute name; and

returning said attribute value.

32. (Previously Presented) The method of claim 27 wherein said producing a fourth implementation further comprises:

receiving a reference name;

producing an implementation that calls a handler method of a superclass of said class, passing said reference name as an argument and returning the reference value associated with said reference name; and

returning said reference value.

33. (Previously Presented) The method of claim 27 wherein said producing a fifth implementation further comprises:

receiving an operation name and any associated arguments;

producing an implementation that calls a handler method of a superclass of said class, passing said operation name and said associated arguments as arguments and returning an operation return value; and

returning said operation return value.

34. (Currently Amended) A program storage device readable by a machine, embodying a program of instructions executable by the machine to perform a method to dynamically

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

implement a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing, dynamically and automatically at runtime, a package proxy JMI interface when said request comprises a package proxy request;

implementing, dynamically and automatically at runtime, a class proxy JMI interface when said request comprises a class proxy request; and

implementing, dynamically and automatically at runtime, a class instance JMI interface when said request comprises a class instance request wherein said implementing operations are performed separately and independently based upon said request.

35. (Original) The program storage device of claim 34 wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface;

creating a new instance of said class; and

returning said instance.

36. (Previously Presented) ~~The program storage device of claim 35~~ A program storage device readable by a machine, embodying a program of instructions executable by the machine to perform a method to dynamically implement a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a package proxy JMI interface when said request comprises a package proxy request wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface wherein said generating further comprises:

receiving a metamodel package;

receiving a package proxy interface method associated with said metamodel package;

determining a class name based upon said interface method;

searching said metamodel package for a class corresponding to said class name; and

producing an implementation of said interface method that returns a proxy for said class when said class name is found in said metamodel package;

creating a new instance of said class; and

returning said instance;

implementing a class proxy JMI interface when said request comprises a class proxy request; and

implementing a class instance JMI interface when said request comprises a class instance request.

37. (Previously Presented) The program storage device of claim 36 wherein said implementation of said interface method calls a handler method of a superclass of said class, passing said class name as an argument and returning the proxy for said class.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

38. (Currently Amended) The program storage device of claim ~~34~~36 wherein said implementing a class proxy JMI interface comprises:

generating bytecode for a class that implements said class proxy JMI interface;

creating a new instance of said class; and

returning said instance.

39. (Previously Presented) The program storage device of claim 38 wherein said generating further comprises:

receiving a metamodel class;

receiving a class proxy interface method associated with said metamodel class;

producing a first implementation of said interface method that creates a new instance of said class when said interface method is parameterless; and

producing a second implementation of said interface method that creates a new instance of said class and sets attributes passed as arguments to said interface method when said interface method includes at least one parameter.

40. (Previously Presented) The program storage device of claim 39 wherein said first implementation calls a handler method of a superclass of said class, passing said class name as an argument and returning a new instance of said class.

41. (Previously Presented) The program storage device of claim 39 wherein said second implementation calls a handler method of a superclass of said class, passing said class name, said attributes, and attribute values as arguments and returning a new instance of said class.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

42. (Currently Amended) The program storage device of claim ~~34~~36 wherein said implementing a class instance JMI interface comprises:

generating bytecode for a class that implements said class instance JMI interface;

creating a new instance of said class; and

returning said instance.

43. (Original) The program storage device of claim 42 wherein said generating further comprises:

receiving a metamodel class;

receiving a class instance interface method associated with said metamodel class, said interface method having an interface method name;

producing a first implementation of said interface method that sets the value of an attribute when said interface method name includes a first prefix and when the attribute associated with said interface method is found in said metamodel class;

producing a second implementation of said interface method that sets the value of a reference when said interface method name includes a first prefix and when the reference associated with said interface method is found in said metamodel class;

producing a third implementation of said interface method that gets the value of an attribute when said interface method name includes a second prefix and when the attribute associated with said interface method is found in said metamodel class;

producing a fourth implementation of said interface method that gets the value of a reference when said interface method name includes a second prefix and when the reference associated with said interface method is found in said metamodel class; and

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

producing a fifth implementation of said interface method that executes an operation when said interface method has the same name as said operation.

44. (Original) The program storage device of claim 43 wherein

said first prefix is "set"; and

said second prefix is "get".

45. (Previously Presented) The program storage device of claim 43 wherein said producing a first implementation further comprises:

receiving an attribute name and an attribute value; and

producing an implementation that calls a handler method of a superclass of said class, passing said attribute name and said attribute value as arguments.

46. (Previously Presented) The program storage device of claim 43 wherein said producing a second implementation further comprises:

receiving a reference name and an reference value; and

producing an implementation that calls a handler method of a superclass of said class, passing said reference name and said reference value as arguments.

47. (Previously Presented) The program storage device of claim 43 wherein said producing a third implementation further comprises:

receiving an attribute name;

producing an implementation that calls a handler method of a superclass of said class, passing said

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

attribute name as an argument and returning the attribute value associated with said attribute name; and

returning said attribute value.

48. (Previously Presented) The program storage device of claim 43 wherein said producing a fourth implementation further comprises:

receiving a reference name;

producing an implementation that calls a handler method of a superclass of said class, passing said reference name as an argument and returning the reference value associated with said reference name; and

returning said reference value.

49. (Previously Presented) The program storage device of claim 43 wherein said producing a fifth implementation further comprises:

receiving an operation name and any associated arguments;

producing an implementation that calls a handler method of a superclass of said class, passing said operation name and said associated arguments as arguments and returning an operation return value; and

returning said operation return value.

50. (Currently Amended) A program storage device readable by a machine, embodying a program of instructions executable by the machine to perform a method to dynamically implement a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

least one class, said at least one class comprising at least one attribute, reference or operation;

implementing, dymanically and automatically, a JMI interface when said JMI interface is unimplemented; and

executing a stored JMI interface implementation when said JMI interface is implemented.

51. (Original) The program storage device of claim 50 wherein

said implementing further comprises:

implementing a package proxy JMI interface when said request comprises a package proxy request and when said package proxy JMI interface is unimplemented;

implementing a class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is unimplemented; and

implementing a class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is unimplemented; and

said executing further comprises:

executing a stored a package proxy JMI interface implementation when said request comprises a package proxy request and when said package proxy JMI interface is implemented;

executing a stored class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is implemented; and

executing a stored class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is implemented.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

52. (Original) The program storage device of claim 51 wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface;

creating a new instance of said class; and

returning said instance.

53. (Currently Amended) ~~The program storage device of claim 52~~ A program storage device readable by a machine, embodying a program of instructions executable by the machine to perform a method to dynamically implement a Java™ Metadata Interface (JMI) to a metamodel, the method comprising:

receiving a JMI implementation request, said request associated with a metamodel, said metamodel comprising at least one package, said at least one package comprising at least one class, said at least one class comprising at least one attribute, reference or operation;

implementing a JMI interface when said JMI interface is unimplemented, wherein said implementing further comprises:

implementing a package proxy JMI interface when said request comprises a package proxy request and when said package proxy JMI interface is unimplemented wherein said implementing a package proxy JMI interface comprises:

generating bytecode for a class that implements said package proxy JMI interface wherein said generating further comprises:

receiving a metamodel package;

receiving a package proxy interface method associated with said metamodel package;

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

determining a class name based upon said interface method;

searching said metamodel package for a class corresponding to said class name; and

producing an implementation of said interface method that returns a proxy for said class when said class name is found in said metamodel package;

creating a new instance of said class; and

returning said instance;

implementing a class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is unimplemented; and

implementing a class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is unimplemented; and

executing a stored JMI interface implementation when said JMI interface is implemented, wherein said executing further comprises:

executing a stored a package proxy JMI interface implementation when said request comprises a package proxy request and when said package proxy JMI interface is implemented;

executing a stored class proxy JMI interface when said request comprises a class proxy request and when said class proxy JMI interface is implemented; and

executing a stored class instance JMI interface when said request comprises a class instance request and when said class instance JMI interface is implemented.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

54. (Previously Presented) The program storage device of claim 53 wherein said implementation of said interface method calls a handler method of a superclass of said class, passing said class name as an argument and returning the proxy for said class.

55. (Currently Amended) The program storage device of claim ~~51~~53 wherein said implementing a class proxy JMI interface comprises:

generating bytecode for a class that implements said class proxy JMI interface;

creating a new instance of said class; and

returning said instance.

56. (Previously Presented) The program storage device of claim 55 wherein said generating further comprises:

receiving a metamodel class;

receiving a class proxy interface method associated with said metamodel class;

producing a first implementation of said interface method that creates a new instance of said class when said interface method is parameterless; and

producing a second implementation of said interface method that creates a new instance of said class and sets attributes passed as arguments to said interface method when said interface method includes at least one parameter.

57. (Previously Presented) The program storage device of claim 56 wherein said first implementation calls a handler method of a superclass of said class, passing said class name as an argument and returning a new instance of said class.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

58. (Previously Presented) The program storage device of claim 56 wherein said second implementation calls a handler method of a superclass of said class, passing said class name, said attributes, and attribute values as arguments and returning a new instance of said class.

59. (Currently Amended) The program storage device of claim ~~51~~53 wherein said implementing a class instance JMI interface comprises:

generating bytecode for a class that implements said class instance JMI interface;

creating a new instance of said class; and

returning said instance.

60. (Original) The program storage device of claim 59 wherein said generating further comprises:

receiving a metamodel class;

receiving a class instance interface method associated with said metamodel class, said interface method having an interface method name;

producing a first implementation of said interface method that sets the value of an attribute when said interface method name includes a first prefix and when the attribute associated with said interface method is found in said metamodel class;

producing a second implementation of said interface method that sets the value of a reference when said interface method name includes a first prefix and when the reference associated with said interface method is found in said metamodel class;

producing a third implementation of said interface method that gets the value of an attribute when said interface method name includes a second prefix and when

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

the attribute associated with said interface method is found in said metamodel class;

producing a fourth implementation of said interface method that gets the value of a reference when said interface method name includes a second prefix and when the reference associated with said interface method is found in said metamodel class; and

producing a fifth implementation of said interface method that executes an operation when said interface method has the same name as said operation.

61. (Original) The program storage device of claim 60 wherein

said first prefix is "set"; and

said second prefix is "get".

62. (Previously Presented) The program storage device of claim 60 wherein said producing a first implementation further comprises:

receiving an attribute name and an attribute value; and

producing an implementation that calls a handler method of a superclass of said class, passing said attribute name and said attribute value as arguments.

63. (Previously Presented) The program storage device of claim 60 wherein said producing a second implementation further comprises:

receiving a reference name and an reference value; and

producing an implementation that calls a handler method of a superclass of said class, passing said reference name and said reference value as arguments.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888

64. (Previously Presented) The program storage device of claim 60 wherein said producing a third implementation further comprises:

receiving an attribute name;

producing an implementation that calls a handler method of a superclass of said class, passing said attribute name as an argument and returning the attribute value associated with said attribute name; and

returning said attribute value.

65. (Previously Presented) The program storage device of claim 60 wherein said producing a fourth implementation further comprises:

receiving a reference name;

producing an implementation that calls a handler method of a superclass of said class, passing said reference name as an argument and returning the reference value associated with said reference name; and

returning said reference value.

66. (Previously Presented) The program storage device of claim 60 wherein said producing a fifth implementation further comprises:

receiving an operation name and any associated arguments;

producing an implementation that calls a handler method of a superclass of said class, passing said operation name and said associated arguments as arguments and returning an operation return value; and

returning said operation return value.

GUNNISON, McKAY & HODGSON, L.L.P.
Garden West Office Plaza
1900 Garden Road, Suite 220
Monterey, CA 93940
(831) 655-0880
Fax (831) 655-0888